



## Blood vessel modeling for interactive simulation of interventional neuroradiology procedures

Erwan Kerrien, Ahmed Yureidini, Jeremie Dequidt, Christian Duriez, René Anxionnat, Stéphane Cotin

### ► To cite this version:

Erwan Kerrien, Ahmed Yureidini, Jeremie Dequidt, Christian Duriez, René Anxionnat, et al.. Blood vessel modeling for interactive simulation of interventional neuroradiology procedures. *Medical Image Analysis*, 2017, 35, pp.685 - 698. 10.1016/j.media.2016.10.003 . hal-01390923

**HAL Id: hal-01390923**

**<https://inria.hal.science/hal-01390923>**

Submitted on 2 Nov 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Blood vessel modeling for interactive simulation of interventional neuroradiology procedures

E. Kerrien<sup>a,c,\*</sup>, A. Yureidini<sup>d,1</sup>, J. Dequidt<sup>a,b</sup>, C. Duriez<sup>a,b</sup>, R. Anxionnat<sup>e,a</sup>, S. Cotin<sup>a</sup>

<sup>a</sup>*Inria, France*

<sup>b</sup>*Univ. Lille, CNRS, Centrale Lille, UMR 9189 - CRISTAL, F-59000 Lille, France*

<sup>c</sup>*Université de Lorraine, Loria, UMR7503, Vandœuvre-lès-Nancy, F-54500, France*

<sup>d</sup>*Amadeus SAS, Sophia-Antipolis, F-06136, France*

<sup>e</sup>*Dept of Interventional Neuroradiology, Univ. Hosp. of Nancy, Nancy, F-54000, France*

---

## Abstract

Endovascular interventions can benefit from interactive simulation in their training phase but also during pre-operative and intra-operative phases if simulation scenarios are based on patient data. A key feature in this context is the ability to extract, from patient images, models of blood vessels that impede neither the realism nor the performance of simulation.

This paper addresses both the segmentation and reconstruction of the vasculature from 3D Rotational Angiography data, and adapted to simulation: An original tracking algorithm is proposed to segment the vessel tree while filtering points extracted at the vessel surface in the vicinity of each point on the centerline; then an automatic procedure is described to reconstruct each local unstructured point set as a skeleton-based implicit surface (*blobby model*). The output of successively applying both algorithms is a new model of vasculature as a tree of local implicit models.

The segmentation algorithm is compared with Multiple Hypothesis Testing (MHT) algorithm [17] on patient data, showing its greater ability to track blood vessels. The reconstruction algorithm is evaluated on both synthetic and patient data and demonstrate its ability to fit points with a subvoxel precision.

Various tests are also reported where our model is used to simulate catheter navigation in interventional neuroradiology. An excellent realism, and much lower computational costs are reported when compared to triangular mesh surface models.

**Keywords:** Blood vessel tracking, Blood vessel reconstruction, Implicit surfaces, Interactive simulation

---

## 1. Introduction

Interest is growing in the potential benefits of computer-based patient-adapted simulators. Simulators offer physicians a safe environment to learn, rehearse and practice. A reduction of time is expected: for training, operation and hospitalization; but also for a more efficient dissemination of

new techniques. Such expectations are heightened in the endovascular treatment of brain aneurysms where highly skilled practitioners handle micro-tools: guidance of catheters and guidewires, and deployment of coils in the aneurysm.

Previous works [10, 60, 49] attest the feasibility of both interactive and physically realistic simulators for such procedures based on patient data such as 3D Rotational Angiography (3DRA). However, appropriate collision information must be computed in order to reach an acceptable level of realism [51], since the simulated medical device is in permanent contact with the vessels wall. In that context, accuracy and performance heavily depend on the blood vessel surface representation.

There mainly exist three surface representations:

---

\*Corresponding author

Email addresses: [erwan.kerrien@inria.fr](mailto:erwan.kerrien@inria.fr) (E. Kerrien), [ahmed.yureidini@gmail.com](mailto:ahmed.yureidini@gmail.com) (A. Yureidini), [jeremie.dequidt@univ-lille1.fr](mailto:jeremie.dequidt@univ-lille1.fr) (J. Dequidt), [christian.duriez@inria.fr](mailto:christian.duriez@inria.fr) (C. Duriez), [r.anxionnat@chu-nancy.fr](mailto:r.anxionnat@chu-nancy.fr) (R. Anxionnat), [stephane.cotin@inria.fr](mailto:stephane.cotin@inria.fr) (S. Cotin)

<sup>1</sup>A. Yureidini participated in this research while at Inria Nancy.

polygonal, parametric and implicit. A surface representation can be extracted from image data through the process of segmentation [32]. Vessel visualization demands a global smooth surface model, accurate enough to visually preserve structures of interest [42, 59]. A different point of view is considered for simulation. Realistic collision requires that appropriate information, such as penetration depth, is retrieved through localized surface queries [51]. Surface representations were discussed regarding collision management in [34]. We briefly review them in consideration of more recent papers and discuss their ability to model vessels for an efficient collision management.

### 1.1. Polygonal models

Polygonal models are very common end-products of segmentation methods. Local shape constraints can be inserted in the segmentation framework, such as tubularity [61]. Triangular mesh surfaces are a natural input to Graphics Processing Units (GPUs) for visualization. Efficient computational structures such as Bounding Volume Hierarchies (BVH) exist for managing collisions between polygonal objects, but hierarchical space representations are not well suited to spatially sparse branching structures. Moreover, handling collisions of deformable objects at interactive rates is still a challenge [51, 30, 25, 8, 35].

### 1.2. Parametric surfaces

Catmull-Clark interpolation proved useful to smooth polygonal vessel surfaces [60, 15] but is not constrained by image data. A popular blood vessel surface model consists in the generalized cylinder parametrization [36, 37, 54]. Graphics primitives were also used to model vascular segments [21]. The vessel section is imposed to be circular, which is not accurate enough for the simulation to be physically realistic, and makes it difficult to model pathological sections and bifurcations.

### 1.3. Implicit surfaces

The Euclidean distance function provides a natural implicit expression for polygonal or parametric models. Sweep surfaces extend the generalized cylinder paradigm and model the vessel cross-section as a 2D spline curve [22, 26]. However, consistency of the estimation in the longitudinal direction is hard to ensure. Convolution surfaces were also proposed for visualization [40]. Though such

surfaces are very smooth, details and other local asymmetries in the shape demand for a complex skeleton to be captured.

Procedural implicit surfaces are of particular interest for simulation. Inclusion/exclusion tests for collision detection can be performed in constant time. Their high order continuity makes it easier to compute friction forces. A rapid segment intersection with the implicit surface can also be achieved via a Newton-Raphson method. Moreover, elegant solutions exist to handle collisions between two procedural implicit surfaces [18].

The Computer Graphics community has devoted a vast literature to the problem of implicit surface reconstruction from data points [2]. Such methods bear no difficulty to interpolate missing data points [41] but handling noise and outlier data points is still considered an issue [38]. Implicit surfaces relying on a point set skeleton have attracted a particular interest for their versatility (capacity to model various shapes), their compacity (low parametrization with regard to the complexity of the shape) and their mathematical simplicity (in particular to compute derivatives in closed form). However, implicit surfaces tend to blend as they come close to each other. As a result, implicit modeling of a complete vasculature is very difficult since, for example, close or kissing vessel branches merge [46]. Controlling the blend is still an open problem when more than two objects are merging or in the case of complex topologies [20].

### 1.4. Vessel centerline tracking

Vascular topology is given by a tree of centerlines augmented with local information on the vessel shape. An impressive set of previous works have been proposed by the Computer Vision community to extract vessel centerlines from 3D angiographic images [32]. Tracking algorithms best fit our context, especially for planning procedures: they provide fast computation times, and the topology of the vascular tree is a direct by-product of such algorithms.

The local tubularity assumption is often leveraged to model blood vessels. A disk can be estimated from image data in a plane estimated to be orthogonal to the vessel direction [55]. The centerline is thereby a tree of polylines formed by the succession of disk centers, each equipped with a radius. However, due to limitations on image quality (noise, artefacts, resolution), the center and radius estimates must be regularized to ensure vessel

smoothness in the longitudinal direction. A representation of vessel branches as a continuous 4D path was proposed in [12, 33]. Centerline paths can also be retrieved by modeling the vessels as local cylinders [27, 57, 58, 45, 17], or elongated blobs e.g. as superellipsoids [53] or through the analysis of the Hessian response [16, 23, 5].

Angiographic data are noisy, and outlier cases to the cylinder shape hypothesis may occur when a vessel gets close to a dense structure (e.g. bone, other vessel, aneurysm) or at bifurcations. The robustness of the image data term was barely addressed in the literature if not for [53] that relied on a probabilistic criterion to separate the object (blood vessels) from the background. Moreover, vessel tracking algorithms most often output only the extracted centerlines as a succession of points (vessel center) usually equipped with the vessel radius. When the vessel surface is required, the inferred reconstruction relies on the disk-shaped section hypothesis. This is too simple in our simulation context where details are important to capture. As an exception, [55] kept the points extracted at the vessel surface while tracking the vessel and triangulated them to visualize the extracted surface. Last, a fixed, user-defined, tracking step was used by previous authors which made handling tortuous and bent vessels difficult, again with the exception of [53] that included the superellipsoid main axis length in the vessel description vector.

## 2. Positioning and contributions

Our objective is the simulation of neuroradiology interventions. Procedural implicit surfaces appear best suited for precise and physically grounded contact computation. But the complex intracranial vasculature is likely to require a complex implicit function, putting interactive simulation rates at risk. Besides, vessel centerlines carry precious topological information to accelerate contact localization and manage unwanted blending. But current implicit vessel representations cannot be used for simulation. Cylinder-like shapes are unable to capture details at the vessel surface, and cross-sectional descriptions suffer from a lack of smoothness in the longitudinal direction. Therefore, we propose a new blood vessel surface model where both concepts are merged in a tree of local implicits. Leveraging the solid literature on implicit surface reconstruction from data points [2], the link

is made through local point sets attached to each vessel center point.

Our work addresses the following challenges. Concerning the vessel centerline extraction, it must be as automatic as possible and produce point sets that are free from any outliers, since robustness is still an issue for implicit reconstruction methods. The RANSAC-Based Tracking (RBT) algorithm (Section 3) runs on a wide variety of cases with the same parametrization, at the cost of a few point clicks, while outliers are filtered out of local surface points. Besides, no current method can fit a procedural implicit surface consistently well enough to handle thousands of cases (one per centerline point) without any manual interaction. The Local Implicit Modeling (LIM) algorithm (Section 4) improves over [52, 3] with a more efficient energy formulation and a more discriminative blob selection criterion.

An extensive validation study demonstrates, on both synthetic and patient data, the strength of our method (Section 5). RBT is proven to outperform the Multiple Hypothesis Tracking (MHT) method [17] on 3D neurovascular images. LIM is shown to fully automatically and accurately fit surfaces to data points, despite holes and noise. Finally, Section 6 shows how collision with the micro-tool can be solved both efficiently and accurately using our model.

## 3. Vessel centerline tracking: The RBT algorithm

### 3.1. Description of the algorithm

The RANSAC-Based Tracking was designed to produce regularized centerlines on both large and small vessels, while being robust to image quality issues, and extract local geometrical features to be used in a second step to reconstruct the vessel surface (see LIM algorithm described in Section 4).

We model the local vessel shape as a finite cylinder  $(C, r, \mathbf{d}, h)$  where  $C$  is the center,  $r$  the radius,  $\mathbf{d}$  the axis direction, and  $h$  the height. Our algorithm alternates between a prediction and an estimation step. State  $t$  is predicted from state  $t - 1$  by:

$$C_t^* = C_{t-1} + s \frac{h_{t-1}}{2} \mathbf{d}_{t-1} \quad (1)$$

and

$$(r_t^*, \mathbf{d}_t^*, h_t^*) = (r_{t-1}, \mathbf{d}_{t-1}, h_{t-1}) \quad (2)$$



where  $s$  is a scaling factor for the height-dependent step size. We set  $s = 0.5$  in order to superimpose successive cylinders.

The estimation step is more complex. First, a set of candidate points is extracted at the vessel surface in the vicinity of the predicted cylinder location. Then the best cylinder  $(C_t, r_t, \mathbf{d}_t, h_t)$  is found together with a subset of fitted points. We propose to add this subset to the vessel shape description. Hereafter, a focus is made on the different steps of the estimation process.

### 3.1.1. Candidate point extraction

In angiographic images, such as 3D Rotational Angiography (3DRA) data, vessels are bright objects on a dark background. The points of minimal directional gradient along rays cast from the cylinder center  $C_t^*$  are collected as candidate points at the local vessel surface (one point per ray).  $N_r$  rays are evenly sampled on the Gaussian sphere to avoid point clustering at the poles, as might happen with spherical coordinates, e.g. using bounded spherical projection images [57]. The prediction step, followed by the candidate point extraction is illustrated in Fig. 1, second from the left.

### 3.1.2. Cylinder fit with RANSAC filtering

Most of the extracted points lie at the local vessel surface, but some points are outliers, due e.g. to neighboring structures or bifurcations. Random Sampling Consensus (RANSAC) is a robust estimation procedure that retrieves the cylinder parameters from a minimal set of points [6]. It relies on a closed-form equation that depends on the parametrization of the cylinder [1]. Parametrizing with the center and radius resorts to fitting a circle to three points in the plane orthogonal to the axis, but it requires the axis direction to be known.

In our case, the axis direction can hardly be estimated from the point normals as in [6]. Indeed, the normals would be computed from noisy image gradients. Therefore, a strategy similar to [17] was adopted:  $N_d$  predefined axis directions are tested, centered on  $\vec{d}_t^*$ ; the best cylinder (center and radius) for each direction is estimated by RANSAC (Fig. 1, second from the right). The  $N_d$  directions are reviewed by increasing angle with  $\vec{d}_t^*$ , and the first valid cylinder is kept. A cylinder is considered valid if its inlier rate is above  $p_{inl}$  and its radius is in  $[r_t^*/2, 3r_t^*/2]$  [57].

In order to cope with a large scale of vessel sizes, the distance threshold for outlier rejection was set

proportional to the estimated radius ( $0.1r_t^*$ ). The set of inlier points is called the consensus set.

### 3.1.3. Refinement and height estimation

The parameters of the returned cylinder are further tuned by minimizing the distance to its consensus set (using Powell's algorithm). Then, the center is adjusted along the axis, to coincide with the median position of the consensus set. Thereby the estimated parameters  $(C_t, r_t, \vec{d}_t)$  are set. A new set of candidate points is extracted: the height  $h_t$  is set to encompass 75% of the inlier points around  $C_t$ . These inlier points  $\mathcal{P}_t$  are returned by the estimation procedure (Fig. 1, right).

RBT produces a representation of the vasculature as a tree where each node is a center point  $C_t$ , with a tuple  $\{r_t, \vec{d}_t, h_t, \mathcal{P}_t\}$ . Each node gives a local picture of the blood vessel around its center.

## 3.2. Implementation details

For each axis direction, a minimum of  $N_{min} = 220$  candidate solutions are systematically explored at the start of each RANSAC procedure. A maximum of  $N_{max} = 500$  trials are run.

If no cylinder with an inlier rate above  $p_{inl}$  was found in any directions, the cylinder with the highest inlier rate over all directions is returned, if this rate is above  $p_{inl}/2$ , else no valid cylinder is returned. This enables to relax the hard inlier threshold and keep the tracking going through seldom image locations where the vessels are less contrasted. Moreover, back-stepping is detected when  $C_t$  lies too close to the formerly tracked vessel centerline segments (distance below  $0.1 r_{t-1}$ ). In such case, the tracking stops.

$N_r$  rays are cast and  $N_p$  points are sampled on each ray along a given length to return the point of minimal directional gradient. Ray casting requires particular care for the algorithm to be efficient. To generate a discrete set of direction axes, the Gaussian sphere was evenly sampled through a recursive subdivision of an icosahedron<sup>2</sup>, leading to  $N_r = 162$  rays for point extraction (number of vertices on the full sphere mesh), and  $N_d = 81$  cylinder axis directions (half sphere).

In [55], a ray was terminated as soon as a gradient norm above a manual threshold was encountered. To avoid early ray termination, e.g. due to local variations of the signal inside the vessel, [57] retain

<sup>2</sup><http://gts.sourceforge.net>

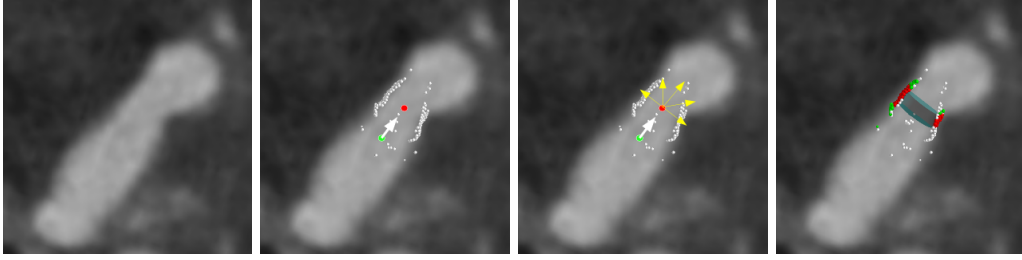


Figure 1: Outline of RBT vessel tracking algorithm (from left to right). 1) 3DRA data (cut plane). 2) Prediction of  $C_t^*$  (red dot) from  $C_{t-1}$  (green dot) and  $\vec{d}_{t-1}$  (white arrow). Candidate points (white dots) are extracted at the vessel surface by ray casting. 3)  $N_d$  cylinder axes are tested (yellow arrows) 4) Best cylinder found ( $C_t, r_t, \vec{d}_t$ ), and its consensus set (green dots). The cylinder height  $h_t$  is fixed using a subset  $\mathcal{P}_t$  (red dots).

the point of minimum directional gradient along a fixed ray length  $\mathcal{L}_t$  (set to  $1.5r_t^*$ ). We found that this failed to capture points at the vessel surface in strongly curved portions due to  $C_t^*$  lying too close to the vessel wall, where  $\mathcal{L}_t > 2r_t^*$  is required, or at bifurcations where the vessel radius can drastically change. We chose the value  $\mathcal{L}_t = 3r_t^*$  which enabled to consistently detect points at the vessel surface.

Various 3D voxel data interpolation schemes were studied. Synthetic experiments aiming at retrieving points at the border of a unit disk in low resolution images, with varying locations for the disk center and ray origin led us to clearly choose ray casting the original volume data using cubic B-spline approximation, followed by gradient computation using central difference in 1D.

### 3.3. Tracking the whole vessel tree

RBT can track a single branch at a time, but it does not stop at bifurcations. The choice for the furcating branch to track depends on its alignment with its parent centerline, its contrast and clarity, but may also be random, depending on RANSAC sampling. RBT proved to be very robust to poor initialization of the cylinder parameters. To add a new branch, the user was asked to manually place a starting point  $C_0^*$ . This initial location was automatically tuned by projecting  $C_0^*$  onto the parent vessel centerline as  $C_0'$ : the initial direction was  $\vec{d}_0 = (C_0^* - C_0')/|C_0^* - C_0'|$  and the initial radius  $r_0$  was set equal to the radius of the parent vessel. The initial point  $C_0$  was set to  $C_0 = C_0' + 1.25r_0\vec{d}_0$ , so as to lie at the start of the new vessel, just outside the parent vessel.

### 3.4. Limitations of RBT: Capturing aneurysms

Aneurysms do not comply well with the cylinder shape prior. Yet, RBT was able to track inside aneurysms, but the extracted topology did not always make sense and surface points went missing. In Fig. 2 the most complex, bilobed, aneurysm in our database was tracked. The centerline was irregular at the junction between the lobes, and made a loop at the end: the last node center should be a neighbor of other centers and a graph topology would be more adapted to the aneurysm shape. Surface points also went missing where humps protrude, as well as at the bottom of the aneurysm sack where the centerline ends.

Instead, a manual procedure was chosen to capture aneurysms, following [55]: center points were placed within the aneurysm (1 or 2 points were enough in most cases). In the absence of a prior shape model, rays were cast from each point, keeping the first location where the directional gradient was above a user-defined threshold. Points already in the consensus set of a vessel centerline were automatically excluded in rare cases (see Kissing Aneurysms (KA) cases in the Validation section 5) to build the corresponding  $\mathcal{P}_t$ . The radius was estimated by fitting a sphere to each point set  $\mathcal{P}_t$ , in the least-squares sense, and center points were linked to complete the vessel centerline. The sub-tree corresponding to the aneurysm was attached to the RBT tree according to 3.3. This procedure worked reasonably well on our data (Fig. 2, right column and Fig. 5). Our current work to better address aneurysms shares the same idea as [29] where a pathology is seen as a deviation from normal vasculature, as RBT can extract it. Isolating the aneurysms from normal vasculature should help segment their surface.

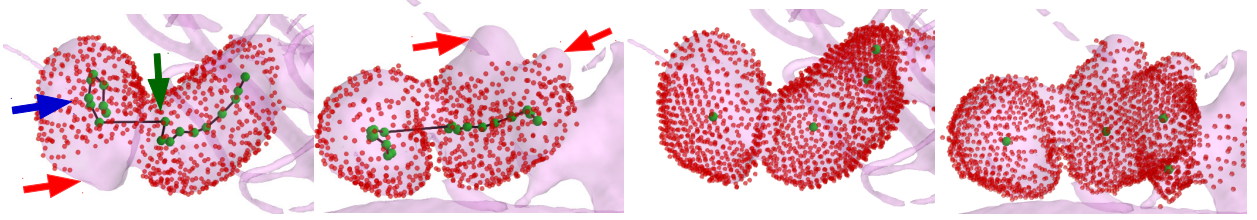


Figure 2: (left) Tracking example inside a bilobed aneurysm (2 different views): node centers (green), and the extracted surface points (red, consensus sets). The extracted centerline is irregular at the connection between the lobes (green arrow) and makes a loop at its end (blue arrow). Points are missing at the surface (red arrows). (right, same 2 views) manually set center points (green) and extracted surface points (red).

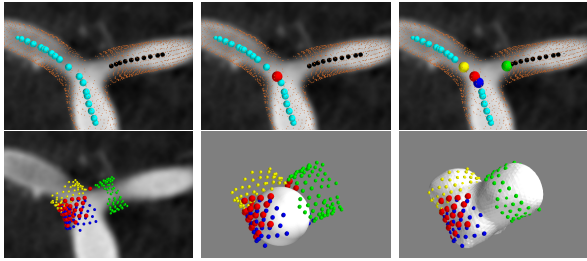


Figure 3: Outline of the LIM algorithm. (top left) Input data (close-up at a bifurcation): RBT node centers (blue and black dots), and surface points (orange dots). (top center) The red node is chosen. (top right) Closest neighbor nodes. (bottom left) Local point sets are concatenated. (bottom center) The local implicit surface (blobby model) is initialized with two close spheres. (bottom right) LIM successively fits and splits blobs to fit the input points (result after 30 subdivisions).

#### 4. Implicit modeling: The LIM algorithm

RBT models the vascular tree as a tree of local point sets, around center points associated with a vessel radius estimate. This depiction is too rough and approximate for an accurate and realistic tool simulation to be conducted. The central idea of local implicit modeling (LIM) is to further enrich each node on this tree with a continuous and smooth representation of the local vessel surface as an implicit surface. Fig. 3 displays the main steps in the reconstruction process for one node in the vicinity of a bifurcation. The same process is applied to each node in the vascular tree.

##### 4.1. Implicit formulation

Implicit surfaces generated by skeletons are particularly well-suited for modeling smooth free-form shapes in a very compact and versatile way [3].

They present good *locality* properties: skeleton objects far apart have no, or at least negligible, influence on each other [9]. In particular, point set skeletons produce procedural implicit functions which are computationally very efficient.

An implicit iso-surface generated by a point-set skeleton [4] is expressed as the zero-level set  $\mathcal{S}$  of a function  $f$ , a sum of  $N_b$  implicit spheres:

$$f(X; p) = \sum_{j=1}^{N_b} \alpha_j \phi \left( \frac{|X - C_j|}{\rho_j} \right) - T \quad (3)$$

where  $X \in \mathbb{R}^3$ ,  $T \in \mathbb{R}^+$  is the iso-surface threshold,  $\{\alpha_j\}$  are positive weights (height), and  $\{C_j\}$  is the point-set skeleton. Each implicit sphere  $\#j$  is defined by a symmetric spherical function called *profile* or *kernel* [47]. The sphere radius is tuned through the width  $\rho_j$  that scales the profile, and its position is set by centering the profile on  $C_j$ .

Such models were called differently depending on the kernel used [47]. Following the seminal work of [39], we shall use the terms *blob* for an implicit sphere, and *Blobby Model* (BM) as a generic name for the implicit models  $f$ . Though our implementation supports several kernels, all our experiments were realized using the Cauchy kernel:  $\phi(x) = (1 + x^2/5)^{-2}$  (we normalized the kernels so that  $\phi''(1) = 0$ ).

In our particular simulation context,  $f(X; p)$  should relate monotonically to the geometric distance of  $X$  to the surface, in order to easily measure interpenetration distance (function value) and direction (function gradient). In the case of a single blob  $\#j$ , the distance to this implicit sphere depends both on  $\alpha_j$  and  $\rho_j$ , but the implicit function returns  $\alpha_j$  at the center, where the distance to the surface is  $\rho_j$ . We therefore set  $\alpha_j = \rho_j$ . Consequently, the implicit function  $f$  is parametrized only by  $p^T = \{C_j^T, \rho_j\}_{j=1 \dots N_b}$  of size  $4N_b$ .

#### 4.2. Energy Formulation

Fitting a surface to  $N_p$  points  $\{P_i\}$  can be written as an energy minimization problem [39, 3, 52]. This energy is made of two parts. First, a data attachment term,

$$\mathcal{E}_d = \frac{1}{N_p} \sum_i f(P_i; p)^2 \quad (4)$$

expresses that data points should lie on the zero-level set. But it does not specify the orientation of the implicit surface. In the case of an oriented point set, the misalignment of the normals with the implicit surface gradient can be minimized through an energy term [39], which can be seen as Poisson problem [24]. We experimentally found such term has no effect in our case where the surface can safely be initialized inside the shape spanned by the point set. Various formulations have been proposed for the other, regularization term. The integral of the gradient norm acts as smoothness term (Tikhonov regularization), but higher-order terms are required to correct for the jaggedness artifacts [31] observed with a sum of implicit spheres. Such terms aim at minimizing the total curvature. The curvature of an implicit function can be computed in a closed form at any point in space [19] as

$$\kappa(P) = \frac{\nabla f^t H_f \nabla f - |\nabla f|^2 \text{trace}(H_f)}{2|\nabla f|^3} \quad (5)$$

where  $\nabla f$  is the implicit function gradient and  $H_f$  its Hessian matrix, both computed at point  $P$ . We therefore used the smoothness term

$$\mathcal{E}_a = \frac{1}{N_p} \sum_i \kappa(P_i)^2 \quad (6)$$

This energy is related to the minimal area criterion. The implicit volume can also be constrained to avoid shrinking [39, 52]. Alternatively, the blob centers can be seen as a particle system that can be prevented from collapsing using a repulsive Gaussian energy function of the inter-particle distance [56] or from dispersion using forces that are repulsive at close range and attractive at long range [48]. Such repulsive-attractive forces can be derived from van der Waals potential, for which a mathematically simple approximation is given by Lennard-Jones potential. But this latter assumes that interacting particles have the same radius. We propose to adapt Lennard-Jones potential to heterogeneous

radii as:

$$\mathcal{E}_c = \frac{1}{N_b(N_b - 1)} \sum_{j \neq k} \left( \frac{s\sqrt{\rho_j \rho_k}}{|C_j - C_k|} \right)^{12} - 2 \left( \frac{s\sqrt{\rho_j \rho_k}}{|C_j - C_k|} \right)^6 \quad (7)$$

Each term in the summation is minimal (with value -1) for  $|C_j - C_k| = s\sqrt{\rho_j \rho_k}$ .  $\mathcal{E}_c$  is repulsive for blobs closer than this distance, and attractive for blobs further away.

We propose to combine these 3 energy terms:  $\mathcal{E} = \mathcal{E}_d + \alpha \mathcal{E}_c + \beta \mathcal{E}_a$  where  $(\alpha, \beta)$  are positive weights. All terms experimentally proved to be complementary. The whole energy is known under a closed-form expression. Closed-form expressions were then derived for its gradients with respect to the BM parameters  $\{\rho_j\}$  and  $\{C_j\}$ .

#### 4.3. Selection-Subdivision

A blob selection and subdivision scheme is followed to reach a compromise in the number of blobs between a good model accuracy and a low complexity (Fig. 4). An exhaustive review of all blobs for selection [39] is time consuming. We experimentally found that relying on local computations of  $\mathcal{E}_d$ , and thereby the algebraic distance to the surface [52], was prone to favor small blobs, thus focusing on details, before dealing with areas roughly approximated by one large blob. Instead, our criterion relies upon the geometric distance approximation proposed by [50] (Fig. 4, middle left). First, we find the point  $P_{i^*}$  farthest to the surface (worst fitted point, Fig. 4, 3rd from the left):

$$i^* = \arg \max_{1 \leq i \leq N_p} \frac{|f(P_i; p)|}{|\nabla f(P_i; p)|} \quad (8)$$

Next, we select the blob  $\#j^*$  whose isosurface is the closest to  $P_{i^*}$  (most influential blob):

$$j^* = \arg \min_{1 \leq j \leq N_b} \frac{|f_j(P_{i^*})|}{|\nabla f_j(P_{i^*})|} \quad (9)$$

with

$$f_j(P_{i^*}) = \rho_j \phi \left( \frac{|P_{i^*} - C_j|^2}{\rho_j^2} \right) \quad (10)$$

The subdivision step then replaces blob  $\#j^*$  with two new ones (Fig. 4, 4th from the left), whose profiles sum up as  $f_{j^*}$ . A small shift (by  $\rho_{j^*}/10$ ) is applied between their centers to avoid infinite values in  $\mathcal{E}_c$ .

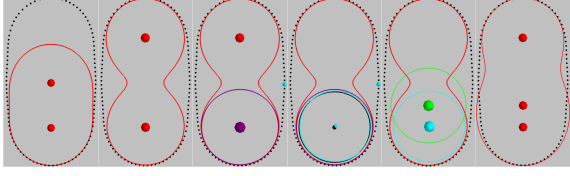


Figure 4: LIM: Elementary step in 2D (from left to right). Data points are in back and the current implicit contour is in red. 1) Initialization with two blobs (centers in red). 2) Result after a first center and width optimization on all blobs. 3) Farthest data point to the implicit surface (light blue) and its closest blob (purple) are found. 4) Closest blob subdivision: one blob is placed at the same location (purple) and the other blob (light blue) is slightly translated towards the farthest point. Widths are computed to preserve the implicit surface (red). 5) Center and width optimization of the new translated blob (light blue). 6) Resulting global implicit surface (red).

#### 4.4. Optimization

The locality of the kernel  $\phi$  allows us to focus the optimization onto the new shifted blob (Fig. 4, 5th from the left). The energy is minimized, alternating between the center (3 variables) and the width (1 variable), using Polak-Ribiere (PR) gradient descent algorithm. The optimization stops as soon as a maximum number of loops or a threshold accuracy  $t_g$  (maximum value found in Eq. 8) is reached. In practice, a maximum of 5 loops proved sufficient. Threshold  $t_g$  will be discussed in Section 5.3.

#### 4.5. Initialization and global modeling

The vascular tree segmentation obtained with RBT is taken as input to our algorithm as a tree of nodes. In order to ensure smooth transitions between adjacent node BMs, the local data point sets  $\mathcal{P}_t$  are concatenated so as to create overlaps between neighbors (Fig. 3, top right). The initial BM is constituted of two blobs placed on either side of the node center  $C_t$  along the vessel direction  $\vec{d}_t$  so that an elongated shape is formed. The width  $\rho_t$  is scaled to match the shape of a sphere with radius  $r_t$  (Fig. 3, bottom center. This match depends on  $\phi$ ). Then, an implicit surface is produced as described above until the maximum distance of the BM to the point set falls below a threshold  $t_g$  or a maximum number  $N_s$  of blobs is reached. Each local BM is treated independently of others, thus enabling parallel computation. Each initial node is enriched with a local BM  $\mathcal{B}_t$ , resulting in a global model of the vasculature as a tree of implicit functions.

Fig. 5 illustrates the capacity that LIM has to model complex shapes, such as aneurysms, even from rough initialization.

### 5. Validation of the segmentation and reconstruction algorithm

Our assessment of the proposed framework was decomposed in three parts. First, we compared RBT against state-of-the-art Multiple Hypothesis Testing (MHT) algorithm [17] on clinical data. Next, the LIM algorithm was tested on synthetic and clinical data. Finally, the effectiveness and performance of model were assessed for the simulation of endovascular navigation. During all our experiments, unless stated otherwise, the RBT and LIM algorithms used a constant parametrization given in table 1.

#### 5.1. Clinical data

We considered 10 patients scheduled for the endovascular treatment of an intracranial aneurysm at the Department of Interventional Neuroradiology, University Hospital of Nancy, France. Each patient data set consisted of a 3D Rotational Angiography (3DRA) acquired on a vascular C-arm (Innova 4100, GE Healthcare) during the intra-arterial injection of the internal carotid artery. 3DRA volumes presented as a  $512^3$  isotropic voxel cubes, with a voxel size between 0.18-0.22 mm.

#### 5.2. Evaluation of our RBT algorithm

##### 5.2.1. Definition of a reference

The Multiple Hypothesis Testing (MHT) algorithm [17] was used as reference for blood vessel segmentation. It achieved the highest score among 13 algorithms, both fully automatic as well as semi-automatic [43]. MHT is available on *MevisLab* software<sup>3</sup> under the *TubularTracking* module.

Multi-branch MHT failed on our data. Consequently, the mono-branch tracking was used. It requires a starting point  $C_0$ , an initial direction  $\vec{d}_0$  and radius estimate  $r_0$ . Section 3.3 explains how  $\vec{d}_0$  and  $r_0$  can automatically be set in RBT but MHT failed under those conditions. As a result, in a view to produce a fair comparison, both MHT and RBT were initialized with a manual setting of  $C_0$ ,  $\vec{d}_0$  and  $r_0$ .

<sup>3</sup><http://www.mevislab.de>

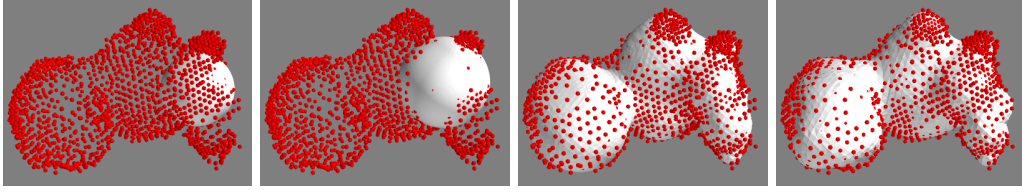


Figure 5: Implicit modeling of an aneurysm. The points  $\{P_i\}$  are in red and the implicit surface in white. (From left to right) Initialization with a single blob; after the first minimization; after 25 subdivisions; final result (100 subdivisions)

<i>RBT</i>			<i>LIM</i>		
Name	Description	Value	Name	Description	Value
$p_{inl}$	inlier rate threshold	70%	$\alpha$	weight of $\mathcal{E}_{LJ}$	$10^{-5}$
$r_t$	threshold relative to cylinder radius	10%	$\beta$	weight of $\mathcal{E}_a$	$10^{-3}$
$N_r$	number of cast rays	162	$s$	controls the repulsive distance in $\mathcal{E}_{LJ}$	2
$N_d$	number of axis direction	81	$T$	iso-level value	0.1
$N_p$	number of points per ray	128	$N_s$	max number of subdivisions	100
$N_{min}$	RANSAC min number of tests	220	$t_g$	targeted point to surface distance	0.3mm
$N_{max}$	RANSAC max number of tests	500			

Table 1: Parameter values for the proposed vessel tracking and reconstruction algorithms.

### 5.2.2. Validation data processing

Both algorithms can track past bifurcations, but they might follow different furcating branches. We therefore defined a *vessel* to be a segment of the vasculature between two bifurcations. Each vessel was further determined by manually setting a final (distal) point  $C_f$ . Each MHT and RBT centerline was automatically clipped to the closest point to  $C_f$  if closer than twice the estimated vessel radius. All resulting centerlines were visually and independently inspected and labeled as: Successful Tracking (ST) if  $C_f$  was reached and no editing was necessary; Premature Stop (PS) if  $C_f$  was not reached but the centerline fully remained inside the targeted vessel; Wrong Tracking (WT) if no centerline was produced, or the parent vessel or a neighbor vessel were tracked right from the start. The remaining cases display a correct tracking of the proximal section of a vessel but fail at some point. Such cases were manually clipped at the first error (Fig. 6 for an example). These edited cases were labeled according to the cause for edition: Kissing Vessel (KV): leakage into a vessel in contact; Kissing Bone (KB): leakage into bone in contact; Kissing Aneurysm (KA): leakage into an aneurysm in contact; Leakage into Background (LB); Back Track (BT): the tracking makes a back step into

the same vessel.

Two experts independently performed the manual steps (setting  $C_f$ , editing and labelling the vessels) and then a consensus was established among them at each step. In 17 cases, one expert missed a bifurcation and the end point was corrected. In all other cases, the experts agreed on the branch path, though not exactly on the position for  $C_f$ : the most proximal position was kept. Carotid arteries sections were excluded from the data set due to their being too large for MHT to track them. Between 57-96 vessels per patient were examined, for a total of 745 vessels.

### 5.2.3. Parametrization of the algorithms

In one patient (Patient 1), 15 vessels of different sizes were randomly selected and manually processed as described above ( $C_0, \vec{d}_0, r_0, C_f$ ). Following a trial and error method, the parameters for MHT and RBT (see Table 1) were tuned so that each algorithm produced a maximal number of ST or PS results.

For MHT, the radius was allowed to vary between 0.25 to 1.25 times the estimated initial radius  $r_0$ . The single hypothesis tracking option was set. The tracking step was set to 0.1 times the current tracking radius, the maximum search angle between 2

tracking steps was  $85^\circ$  and the number of search angles was 5. A maximum of 2000 iterations were allowed and all other parameters initialized to default values.

#### 5.2.4. Qualitative and quantitative assessment

A first set of measures helped assessing the general ability to track vessels, and a second aimed at quantifying the accuracy of RBT centerlines with respect to MHT. These measures are similar to those defined in [44], but had to be tweaked to deal with a (at times imperfect) reference method, instead of a ground truth.

*Assessing the tracking ability.* Successes are given by ST and PS labeled centerlines while WT are failures. Edited centerlines can be considered either way, depending on whether manual edition is allowable or not in the clinical setup. Besides label distribution as percentages, the cumulative length of the centerlines was computed without and with inclusion of the edited cases (respectively denoted TL (Tracked Length) and TLE (TL with edition)).

*Accuracy.* We assessed the distance between two centerlines with a Modified Average Symmetrical Distance (MASD):

$$\text{MASD} = \frac{1}{N_w + M_w} \left( \sum_i^N d_w(P_i, \mathcal{Q}) + \sum_j^M d_w(Q_j, \mathcal{P}) \right) \quad (11)$$

where  $\mathcal{P} = \{P_i\}_{i \in [1, N]}$  is the centerline produced by MHT, and  $\mathcal{Q} = \{Q_j\}_{j \in [1, M]}$  is produced by RBT.  $d_w$  is the classical Euclidean distance between a point and a curve, modified to return 0 whenever it exceeds  $w$ .  $N_w$  and  $M_w$  are the number of points respectively in  $\mathcal{P}$  and  $\mathcal{Q}$  where the distance was not thresholded. MASD automatically selects the portions common to both centerlines and computes the mean distance between them (Fig. 6, second row).

Two values were used for  $w$ :  $w = 3$  voxels and  $w = \bar{r}$ , the average radius of the tracked vessel. Similarly low results would prove the capacity of the measure to consistently select the correct common portions of the vessel centerlines.

#### 5.2.5. Results and discussion

Table 2 reports label percentages for MHT and RBT. The number of WT is divided by 2 with RBT (4.16% vs 9.66%). Besides, RBT shows an increase of ST results by 43% over MHT (82.55%

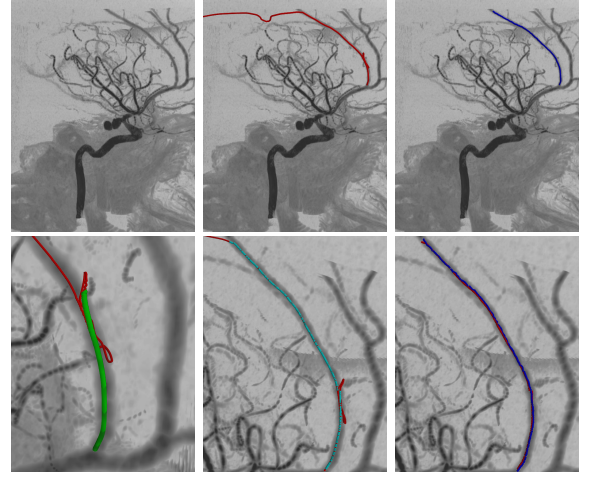


Figure 6: Top: (left) Maximum Intensity Projection of one patient data: profile view of the left internal carotid artery. Centerline produced by MHT (middle) and RBT (right). Bottom: (left) MHT centerline (red) steps out of the vessel: it was labeled BT and cut for TLE computation (green). (middle) MHT centerline (red) and the points (cyan) selected for MASD computation ( $w = \bar{r}$ ). (right) The resulting points for MHT (red) and RBT (blue, all the original points) used for MASD computation.

against 59.87%) and even 58.7% if we include PS results (95.44% vs 60.14%). In this latter case, a TL of 10909.5 mm was measured for RBT, against 5959 mm (83% increase). This first result clearly demonstrates the greater tracking ability of RBT over MHT. This is confirmed even when we include edited cases. These cases are anecdotal for RBT (3 KV cases) while they represent 30.2% of MHT results, half of them being due to a perturbation by a neighboring dense structure. TLE was measured 11637.7 mm for RBT against 8407.8 for MHT (38.4% increase) with only a 5% difference in the number of cases considered (95.84% vs 90.34%). The presence of dense neighboring structures (KV, KB and KA labels) clearly perturbs MHT while RBT is more immune to them (Fig. 7 gives an example). Indeed, RBT fits a cylinder shape to the data using RANSAC which is robust to outliers, contrary to the weighted least squares criterion used by MHT.

The average MASD between both centerlines, with both 3 voxels and the mean radius of the targeted vessel, was 0.18 mm (down to 2 figures of precision) and was below one voxel for all patients (min=0.16 mm, max=0.21 mm with a voxel size between 0.18 and 0.22 mm in our patient data set). This showed that RBT produced similar results as

	ST	PS	WT	KV	KB	KA	LB	BT
MHT	59.87	0.27	9.66	13.02	2.55	0.27	8.86	5.50
RBT	82.55	12.89	4.16	0.40	0.00	0.00	0.00	0.00

Table 2: Percentage distribution of labels assigned to centerlines extracted by MHT and RBT: ST=Successful Tracking, PS=Premature Stop, WT=Wrong Tracking, KV=Kissing Vessel, KB=Kissing Bone, KA=Kissing Aneurysm, LB=Leakage into Background, BT=Back Turn (see text for details).

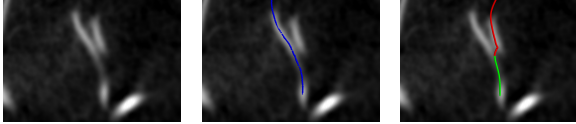


Figure 7: Example of KV occurrence. (left) 3D data: cut plane through a branch of the middle cerebral artery. (middle) RBT successfully tracked the vessel. (right) MHT was misled by a kissing vessel (red) and labeled KV. The green part was kept after manual edition.

MHT.

### 5.3. Evaluation of our LIM algorithm

#### 5.3.1. Parametrization of LIM

After applying RBT to the carotid artery of one patient, one node of the centerline was considered. The energy weights were tuned to provided energy terms of the same order of magnitude. The isolevel value  $T$  was chosen to be approximately half the voxel size. Other values were chosen to produce visually good results. In particular, for Lennard-Jones energy,  $s$  was set to 2: Two blobs with the same width  $\rho$  are thereby separated by  $2\rho$  at this energy minimum. Local point sets were concatenated over a topological distance of 2 (Section 4.5). Values are summarized in Table 1 and were used for all synthetic and patient tests.

#### 5.3.2. Synthetic experiment

*Data.* Four synthetic shapes were used in a first experiment. They consist of skeleton-based convolution surfaces of width approximately 1 mm (blue shapes on Fig. 8): a capsule (**C**, 4 cm segment skeleton), an open capsule (**oC**, based on **C**), an arc (**A**, half-circle of 4 cm radius), a bifurcation (**B**, two arc portions). LIM algorithm was run using 300 points (called  $\mathcal{P}_t$  in section 4.5) sampled on each shape [56].

*Protocol.* Each point in  $\mathcal{P}_t$  was perturbed by a white Gaussian noise along the normal to the reference surface. The standard deviation of the noise

varied from 0.025 to 0.25 mm (step 0.025, 10 levels of noise). LIM was applied 30 times for each noise level, with  $t_g$  set to 3 times the standard deviation.  $N_s$  was set to a large value (100) in order to evaluate the capacity of LIM to produce compact models (i.e. BMs with a reduced number of blobs). LIM initial BM contained 2 (**C** and **oC**) or 5 (**A** and **B**) blobs. The accuracy of LIM was evaluated by robust statistics on the Euclidean distance between  $\mathcal{P}_t$  (noise free version) and the resulting blobby model: The output BM was sampled on 1000 points (again using [56]) to compute the distance of each point in  $\mathcal{P}_t$  to the BM and the 90th percentile was taken: this score is denoted  $d$ . The number of blobs in the output BM was also recorded.

LIM relies on the distance approximation to an implicit surface proposed by [50]. We took here the opportunity to evaluate the adequacy of this approximation and computed it together with the Euclidean distance. The 90th percentile is also reported, and denoted  $d_a$ .

*Result.* Table 3 reports the minimum, median and maximum of both  $d$  and  $d_a$  for the 300 tests, as well as the number of blobs in the final BM.

Excellent distance figures were recorded, with a low median value close to the minimum and a maximum value below the maximum standard deviation of the applied noise. A higher median value was observed for the open capsule **oC** which exposes a more challenging fitting problem due to its open extremities. Fig. 8 displays LIM output for the recorded maximum  $d$  value. We can see that the points at maximum distance are located in the vicinity of the open extremities. This issue is tackled in our case by letting successive BMs overlap (see Section 4.5).

The median number of blobs in the final BMs varied between 7 (**C**) and 15 (**B**), which demonstrates the capacity of LIM to produce compact models. Overly large BMs were seldom obtained when the noise generation process randomly applied large displacements to some points. LIM is indeed not ro-



Shape		C	oC	A	B
$d$	min	0.015	0.039	0.016	0.048
	med	0.042	0.119	0.063	0.078
	max	0.189	0.233	0.174	0.334
$d_a$	min	0.015	0.038	0.016	0.046
	med	0.041	0.103	0.064	0.079
	max	0.164	0.175	0.146	0.211
$N_b$	min	5	5	8	12
	med	7	9	15	15
	max	89	83	102	15

Table 3: Evaluation of LIM accuracy and compacity on 4 synthetic shapes (capsule C, open capsule oC, arc A and bifurcation B) perturbed by a Gaussian noise.  $d$  is the 90th percentile of the Euclidean distance of noise free points in  $\mathcal{P}_t$  to the output BM.  $d_a$  presents the same statistics using the approximate distance proposed by [50].  $N_b$  is the number of blobs in the output BM. Minimum (min), median (med) and maximum (max) values are reported over 300 tests.

bust but the filtering task is devoted to RBT.

The approximate distance  $d_a$  gave very similar results to  $d$ .

### 5.3.3. Reconstruction on patient data

Input data was provided by RBT algorithm which was first run on the carotid artery and subsequently, all vessels were tracked (see Section 3.3). A total of 379 instances of RBT were needed (26-56 instances per patient). Moreover, a total of 11 aneurysms were segmented with the algorithm described in Section 3.4. A total of 87564 BMs were to be fitted by LIM on these patient data.

The accuracy however depends on two parameters that required more care to be fixed.  $N_s$  is the maximum number of subdivisions: the larger  $N_s$ , the more blobs, allowing for greater modeling versatility;  $t_g$  is the maximum distance error: this error is used to determine  $i^*$  (Eq. 8). We tested 3 values for both  $N_s$  ( $\{30, 50, 100\}$ ) and  $t_g$  ( $\{0.2, 0.3, 0.5\}$ , in mm). The resulting 9 configurations were evaluated based on precision and compacity of the output model.

*Visual assessment.* Local BMs generated by LIM do not assemble into a representation suitable to the visualization of the global vessel tree. A rather good approximation can however be given by first meshing each local BM (we used marching cubes), then clipping each local surface within its cell in the inferred power diagram (see Section 6.2.1 for a

precise definition). Finally all these local surfaces are visualized together for a given BM tree. An example is given in Fig. 9.

LIM was able to reconstruct both tiny and large vessels, and capture well even tortuous vessels. Even though each BM is processed independently from its neighbors, overlapping neighboring point sets produced smooth transitions between BMs. A visual comparison with a classical isosurface extraction revealed that more vessels are visible with our proposal (Fig. 9, second to left). Kissing vessels were handled well and no unwanted blending occurred (Fig. 9, second to right). Bifurcations were correctly handled. Some convergence issues were recorded in some places where a very small vessel branches onto a large vessel (e.g. the anterior choroidal artery on the internal carotid artery, Fig. 9, right).

*Precision.* In the absence of ground truth, we resorted to the statistics on the distance approximation called  $d_a$  in Section 5.3.2. Our synthetic experiment indicated  $d_a$  was a good estimate of the score  $d$  based on the Euclidean distance.

Blobby Model outputs (87654) were separated into 4 classes based on their  $d_a$  score:  $C_1 \equiv d_a < 0.5 v_s$ ,  $C_2 \equiv 0.5 v_s \leq d_a < v_s$ ,  $C_3 \equiv v_s \leq d_a < 2 v_s$  and  $C_4 \equiv 2 v_s \leq d_a$ . The voxel size  $v_s$  varies between 0.18 and 0.22 mm per patient and is related to the point extraction noise in RBT. Table 4 reports the distribution, in percents, of  $d_a$  among the 4 classes.

99% of BMs present subvoxel  $d_a$  ( $C_1 \cup C_2$ ) for all configurations, thereby demonstrating that LIM can fit points with a subvoxel precision.  $N_s$  does not appear to have any influence on the precision of fit, which is governed by  $t_g$ . Targeting a precision of  $t_g = 0.2$  mm led to less smooth models where LIM adjusted to noise in the input point set.  $t_g = 0.3$  mm was the best compromise between smoothness and precision.

*Compacity.* A measure of compacity is given by the number of blobs in a BM. A first measure of compacity was the total number of blobs in the BMs required to model a patient’s vasculature. We found that this number did not depend on  $N_s$  and we hereby report only figures for  $N_s = 100$ . The average number of blobs per patient was: 33930 ( $t_g = 0.5$  mm, average of 3.9 blobs per BM), 56395 ( $t_g = 0.3$  mm, average of 6.4 blobs per BM), and

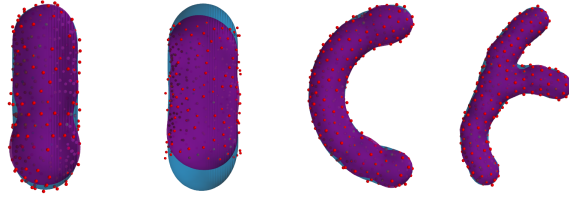


Figure 8: Reference surface (blue) and the resulting BM (purple) for maximal values in Table 3. The 300 noisy input points are in red. (from left to right) capsule, open capsule (notice there is no input point covering the extremities), arc and bifurcation.

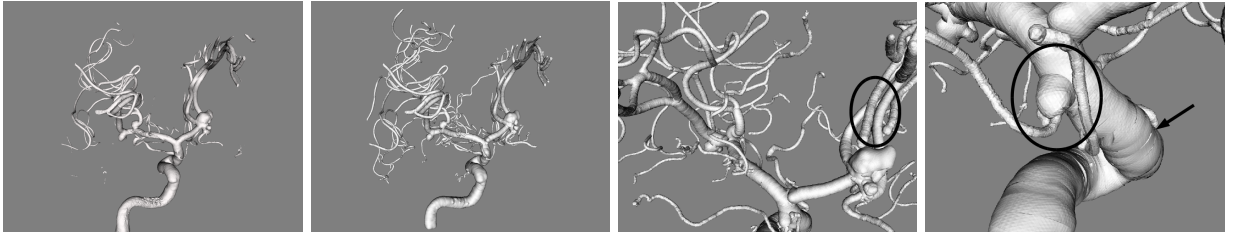


Figure 9: Visual assessment of LIM: (left) isosurface of 3DRA data; (second to left) LIM result, showing much more vessels; (second to right) no unwanted blending at kissing vessels (encircled); (right) difficulty to sometimes model branching of a tiny vessel on a large one (encircled), small discontinuities are visible between adjacent BMs (arrow).

85733 ( $t_g = 0.2$  mm, average of 9.5 blobs per patient).

The number of blobs is the number of primitives required to encode the implicit surface. In the case of a triangular mesh, a similar measure is given by the number of triangles. Therefore, we computed for each patient the ratio between the number of triangles in a mesh representation of the surface, and the total number of blobs in the BMs used to implicitly model the whole vasculature. It indicates the ability of LIM to compress the surface representation. Results are reported in Table 5. For a targeted precision  $t_g = 0.3$  mm ( $N_s = 100$ ), the expected compression ratio is over 100.

## 6. Implementation in a simulation framework

### 6.1. Application context and simulation framework

Our primary application context is interventional neuroradiology and in particular the endovascular occlusion of intracranial aneurysms. Interventional tools are catheters, guide-wires and coils. They all have the shape of a flexible wire that the neurointerventionist pushes, pulls and twists to navigate through the patient’s vasculature. To simulate the

tool, its shape is obtained at the equilibrium between internal forces and contact forces. The former are encapsulated in a FEM model as a set of serially linked beam elements [7]. The latter stem from constraints given by Signorini’s contact laws and Coulomb’s friction laws [13]. This mechanical model was implemented in the real time simulation framework Sofa [14].

### 6.2. Contact management with LIM models

In the simulation loop, contacts are managed by reviewing each point on the wire-like tool for possible collision and solving each detected contact [11].

#### 6.2.1. Detecting collisions

Collision with an implicit surface is easily detected through the sign of the implicit function: in our case, a point crossed the surface if  $f$ , the local BM function, became negative (Eq. 3). For each point on the tool, a node must be selected in the RBT+LIM tree to determine the local BM (see Section 4.5). The node, whose center is the closest to the colliding point, is not appropriate. First, using the globally closest node, the point could occasionally jump from one artery to another one, between two time steps, e.g. in the case of kissing vessels, which is mechanically impossible. Therefore, after

$t_g$	0.5mm			0.3mm			0.2mm		
$N_s$	100	50	30	100	50	30	100	50	30
$C_1$	51.37	51.37	51.37	88.65	88.63	88.54	97.15	96.96	96.58
$C_2$	47.67	47.64	47.61	11.00	10.96	10.97	2.60	2.73	3.01
$C_3$	0.84	0.85	0.86	0.25	0.28	0.34	0.15	0.20	0.26
$C_4$	0.12	0.14	0.16	0.10	0.12	0.15	0.09	0.11	0.14

Table 4: Distribution in % of the BMs according to the error of fit  $d_a$  in 4 classes (left column):  $C_1$  corresponds to BMs with  $d_a < 0.5 v_s$ ,  $C_2$  to  $0.5 v_s \leq d_a < v_s$ ,  $C_3$  to  $v_s \leq d_a < 2 v_s$  and  $C_4$  to  $2 v_s \leq d_{bm}$ .  $v_s$  is the voxel size. BMs in  $C_1 \cup C_2$  were fitted with subvoxel precision. 9 algorithm configurations were investigated depending on parameters  $t_g$  (targeted precision of fit) and  $N_s$  (maximum model complexity). Computations implied 87564 BMs.

$t_g$ (mm)	$\frac{\#\Delta}{N_b}$		
	min	med	max
0.5	159	234	308
0.3	73	104	138
0.2	44	58	77

Table 5: Ratio between the number of triangles  $\#\Delta$  in a triangulated surface and the number of blobs  $N_b$  in the local implicit model for 3 values of  $t_g$  ( $N_s = 100$ ). Minimum (min), median (med) and maximum (max) values are reported for 10 patients.

motion, a point is associated to either the same node as before, or one of its topological neighbors in the tree (Fig. 10).

Second, selecting the closest node resorts to finding the cell the point belongs to in the Voronoi diagram of the node centers. This strategy fails where a small vessel branches onto a larger one: The BM modeling the proximal portion of the small vessel might be selected while the tool point is still inside the large vessel. Using a power diagram enables to adapt the size of the cell to the estimated vessel radius  $r$  at the node. This diagram is generated with the power pseudo-distance :

$$d^p(X, C) = |X - C|^2 - r^2 \quad (12)$$

where  $X$  is the tool point,  $C$  the node center and  $r$  the estimated vessel radius at the node. After motion, the point is associated with its closest candidate node according to  $d^p$ .

At initialization, all the nodes in the tree are viable candidates for the proximal point of the tool. The closest node (in  $d^p$  sense) is selected for this point. Then, each point is successively associated with its  $d^p$ -closest node, among the neighbors of the node selected for the previous point.

### 6.2.2. Solving contacts

The implicit expression of the surface allows for a fast projection of the colliding point  $X$  back onto the surface as  $X'$  with a few Newton-Raphson iterations.  $XX'$  defines the interpenetration vector that gives the contact response force to obey Signorini's law of contact. Note that one Newton-Raphson step moves the point backward by Taubin's approximate distance  $|X|_a = \frac{f(X)}{|\nabla f(X)|}$  along the unit gradient

vector  $\frac{\nabla f(X)}{|\nabla f(X)|}$ . Its computation is fast since the implicit function  $f$  and its gradient are known in a closed form at any point  $X$  in space.

The same distance can be leveraged to search for the point of contact. Let  $X_t$  be the position of the tool point, inside the vessel, at time step  $t$ , and  $X_{t+1}$  be the same tool point, outside the vessel, at next time step. The false position method was applied to find the zero of  $g(\lambda) = |(1 - \lambda)X_t + \lambda X_{t+1}|_a$ , which gives the point of contact.

## 6.3. Evaluation of the vessel model for interactive simulation

### 6.3.1. Evaluation criteria and surface data

We evaluated the impact of our model on simulation with regard to 3 expected benefits: faster collision management, improved realism, and improved numerical stability. The evaluation compared simulations of the navigation of a wire-like tool within various shapes represented implicitly by our model and by triangular meshes. The time to manage a collision was recorded and the navigation sequences were visually inspected.

*Synthetic data.* A *capsule* (Fig. 11 left) was synthesized from a 10 mm line segment skeleton sampled on 10 points, with a radius of 1.25 mm. It provides the most simple vessel shape. A *spiral* (Fig. 11

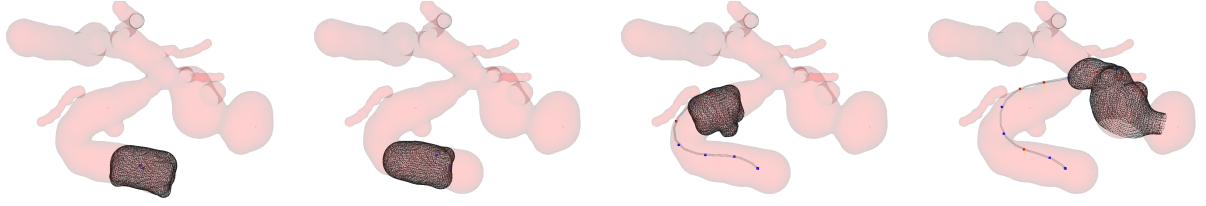


Figure 10: The surface of the local BM selected to solve contact at the tool tip point is displayed in wireframe for 4 simulation steps. The overall vessel surface is shown in transparent red as a mesh. The first three images display steps during catheter navigation in the internal carotid artery and the rightmost image shows the selected BM when the micro-tool enters the aneurysm.



Figure 11: Data used for validation: capsule (left) and spiral (middle) surfaces and (right) the silicone vascular model. Both synthetic shapes presented respectively dimensions (width×height×depth) in mm:  $11.9 \times 2.5 \times 2.5$  and  $8.1 \times 6.12 \times 1.05$ . A primitive was placed at each red dot composing the skeleton (white).

middle) was generated from 100 points on a curve line with polar equation  $(\rho, \theta)$  s.t.  $\rho = \frac{\lambda\pi}{\theta^2}$  with  $\lambda \in [15.75, 63]$  and  $\theta \in [\pi, 4\pi]$ . The shape radius varied from 0.15 mm to 0.5 mm. A smooth surface was obtained in both cases as a convolution surface using Cauchy kernel.

The Marching Cubes algorithm was used to generate a detailed triangular mesh (12k triangles) of the capsule, and 3 triangular meshes (2.74k, 11.3k and 45k triangles) for the spiral. A Newton-Raphson projection technique along the normal to the convolution surface was carried out on the mesh vertices so that they laid in average less than  $10^{-7}$  mm away from the surface.

A similar mesh was used to initialize a dynamic system with 3000 particles that were optimized to regularly sample each shape [56]. The skeleton centers were used as center nodes to simulate the RBT output. Input RBT point sets were built by linking each particle to its closest center. The radius for each node was set to the smallest distance between a point set and the associated center node.

*In-vitro data.* A 3DRA volume was acquired on a C-arm (Innova 4100, GE Healthcare) of a silicone vascular model with 3 aneurysms (Elastrat, Geneva) (Fig. 11, right). Our RBT+LIM modeling algorithm was applied to the  $512^3$  voxel data (voxel size of 0.2 mm), resulting in 558 nodes and a total of 9.5k blobs.

A triangular mesh was initialized with the Marching Cubes algorithm and refined with an active surface model [28]. The resulting triangulated surface was composed of 100k triangles.

### 6.3.2. Endovascular embolization simulation setup

The wire-like tool was modeled with 200 beam elements. Young’s modulus of tools used in interventional radiology ranges from 0.05 to 100 GPa (catheters are mainly composed of silicone rubber, and guidewires and coils are both composed of titanium or alloys). We chose a very stiff medical device of 100 GPa for Young’s Modulus and 0.48 for Poisson’s ratio. The time step was fixed to 1 ms for all simulations. Constant speed (50 mm/s) and direction were applied to the proximal beam element to simulate user input.

The collision detection process with meshes was based on state-of-the-art techniques: broad phase spatial partitioning and axis-aligned bounding box hierarchies for the narrow phase (see [51] for an overview of collision detection techniques with meshes).

### 6.4. Collision management time

The average time taken to detect and solve one collision was measured both for implicit LIM and mesh models. Results are reported in Table 6. Collision management is significantly faster with LIM than with triangular meshes (about two orders of

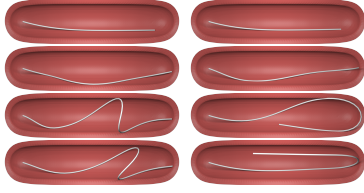


Figure 12: Simulation of catheter deployment inside a capsule-like shape. The collision is handled with a triangular mesh (left) and LIM (right). Even with a fine resolution mesh (12k triangles) artificial friction appears that eventually creates a loop in the catheter. LIM (12k blobs) enables the catheter to smoothly slide along the surface.

magnitude). In fact, collision management time does not depend on the total number of primitives for LIM: the capsule have 12k blobs and the spiral 28k blobs but they display similar computation times. This time only depends on the number of blobs in the locally selected BM, at most a few tens whatever the complexity of the global shape. BM selection is also done in constant time since we restrict the search to topological neighbors of the current BM.

#### 6.5. Simulation realism and numerical robustness

Figure 12 displays steps in the navigation of the tool in the capsule. Even though contacts in the case of the mesh (left) are frictionless, the tip of the device gets stuck due to artificial friction induced by the approximation of the contact surface as a union of planes. This drawback can be attenuated by increasing the mesh resolution but it significantly increases the time to perform the collision detection as well as the resolution of the system. The behavior of the tool is much more realistic and smooth with the capsule modeled with LIM and some level of friction put into the process (right).

When objects move fast, the solver needs more iterations to solve contacts. In a time-constrained real-time simulation framework, the maximum number of iterations must be clamped. At times, the contact resolution error is above the acceptable threshold for the resolution at next time step to be correctly initialized. For example, our device (Young’s modulus=100 GPa) requires more than 200 iterations per time step. Because of the larger contact resolution time triangular meshes require, 100 iterations were allowed at most for the simulation time step to stay below 20 msec. This made the simulation diverge, resulting in the device penetrating the contact object surface. This occurred

with very stiff devices in the capsule, but was recurrent in the spiral even with softer devices and slower motions. LIM enables faster contact response computation rates, allowing for a larger number of iterations. Also, the gradient of the implicit function is defined continuously and smoothly anywhere in space, offering a more consistent definition of the contact force that helps the solver to recover from a non-optimal contact resolution in the previous time step.

## 7. Conclusion

We proposed a new framework to segment a vasculature, via tracking using RBT, and reconstruct the vessel surface, as a tree of local implicit functions using LIM. It combines the benefits of both an implicit surface representation, for its smoothness and modeling versatility, and the knowledge of the vessel tree structure, to avoid unwanted blending and help device navigation.

RBT was proven to be able to track more vessels and on greater lengths than the state-of-the-art MHT method. In particular, RBT is more robust to the kissing vessel issue. The centerlines extracted by both algorithms were shown to have similar precision. RBT produces a richer result than former tracking algorithms: Candidate points are also picked at the vessel surface using a robust filtering approach. These points offer a closer depiction of the complexity of the local vessel surface, than the classical circle or ellipse used for the axial vessel shape.

LIM was shown to generate implicit surface models to fit these point sets with a subvoxel accuracy, despite the presence of holes (e.g. at bifurcations) or irregular sampling density observed in such point sets. LIM expresses the fit problem as an energy minimization approach. Heuristics were described for a progressive select and divide strategy to successfully converge towards stable and compact blobby models. In particular, using Taubin’s approximate distance function proved to be a better choice than the algebraic distance. The closed-form expressions of the objective function and its gradient enabled to leverage efficient gradient-based algorithms for the energy minimization.

Finally, our implicit model was used to simulate interventional radiology procedures. The performance increase, both in terms of computation time and realism, was demonstrated on both synthetic and real examples.

	# of primitives	Time (msec)	# of contacts
Capsule (Mesh)	12k triangles	4.89	51
Capsule (LIM)	12k blobs	0.0232	50
Spiral (Mesh)	2.74k triangles	1.09	79
Spiral (Mesh)	11.3k triangles	1.84	80
Spiral (Mesh)	45k triangles	5.58	80
Spiral (LIM)	28k blobs	0.0219	80
Vascular Model (Mesh)	100k triangles	16.23	80
Vascular Model (LIM)	42k blobs	0.133	82

Table 6: Computation timings for collision detection on various deployment scenarios, using state of the art implementations. For a comparable amount of primitives, LIM outperforms triangular meshes for collision detection and makes it easier to achieve interactive simulation.

## References

- [1] Beder, C., Förstner, W., 2006. Direct solutions for computing cylinders from minimal sets of points, in: 9th European Conference on Computer Vision (ECCV’06), pp. 135–146. doi:10.1007/11744023\_11.
- [2] Berger, M., Levine, J.A., Nonato, L.G., Taubin, G., Silva, C.T., 2013. A benchmark for surface reconstruction. *ACM Transactions on Graphics* 32, 20:1–20:17. doi:10.1145/2451236.2451246.
- [3] Bittar, E., Tsingos, N., Gascuel, M.P., 1995. Automatic reconstruction of unstructured 3d data: Combining a medial axis and implicit surfaces. *Computer Graphics Forum* 14, 457–468. URL: [http://dx.doi.org/10.1111/j.1467-8659.1995.cgf143\\_0457.x](http://dx.doi.org/10.1111/j.1467-8659.1995.cgf143_0457.x), doi:10.1111/j.1467-8659.1995.cgf143\_0457.x.
- [4] Blinn, J.F., 1982. A generalization of algebraic surface drawing. *ACM Trans. Graph.* 1, 235–256. doi:10.1145/357306.357310.
- [5] Carrillo, J., Hoyos, M., Dávila, E., Orkisz, M., 2007. Recursive tracking of vascular tree axes in 3D medical images. *International Journal of Computer Assisted Radiology and Surgery* 1, 331–339. doi:10.1007/s11548-007-0068-6.
- [6] Chaperon, T., Goulette, F., 2001. Extracting cylinders in full 3D data using a random sampling method and the Gaussian image, in: *Vision Modeling and Visualization Conference (VMW’01)*, pp. 35 – 42.
- [7] Cotin, S., Duriez, C., Lenoir, J., Neumann, P., Dawson, S., 2005. New approaches to catheter navigation for interventional radiology simulation, in: Duncan, J., Gerig, G. (Eds.), *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2005*. Springer Berlin Heidelberg, volume 3750 of *Lecture Notes in Computer Science*, pp. 534–542. doi:10.1007/11566489\_66.
- [8] Curtis, S., Tamstorf, R., Manocha, D., 2008. Fast collision detection for deformable models using representative-triangles, in: *Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games*, ACM, New York, NY, USA, pp. 61–69. doi:10.1145/1342250.1342260.
- [9] Dekkers, D., van Overveld, K., Golsteijn, R., 2004. Combining CSG modeling with soft blending using Lipschitz-based implicit surfaces. *The Visual Computer* 20, 380–391. doi:10.1007/s00371-002-0198-3.
- [10] Dequidt, J., Duriez, C., Cotin, S., et al., 2009. Towards interactive planning of coil embolization in brain aneurysms, in: *MICCAI 2009*. Springer-Verlag, volume 5761 of *LNCS*, pp. 377–385. doi:10.1007/978-3-642-04268-3\_47.
- [11] Dequidt, J., Lenoir, J., Cotin, S., 2007. Interactive contacts resolution using smooth surface representation, in: *Proceedings of the 10th international conference on Medical image computing and computer-assisted intervention*, Springer-Verlag, Berlin, Heidelberg, pp. 850–857.
- [12] Deschamps, T., Cohen, L., 2002. Fast extraction of tubular and tree 3D surfaces with front propagation methods, in: *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, pp. 731–734 vol.1. doi:10.1109/ICPR.2002.1044862.
- [13] Duriez, C., Andriot, C., Kheddar, A., 2004. Signorini’s contact model for deformable objects in haptic simulations, in: *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, IEEE, pp. 3232–3237.
- [14] Faure, F., Duriez, C., Delingette, H., Allard, J., Gilles, B., Marchesseau, S., Talbot, H., Courtecuisse, H., Bousquet, G., Peterlik, I., Cotin, S., 2012. SOFA: A Multi-Model Framework for Interactive Physical Simulation, in: Payan, Y. (Ed.), *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*. Springer, volume 11 of *Studies in Mechanobiology, Tissue Engineering and Biomaterials*, pp. 283–321. URL: <https://hal.inria.fr/hal-00681539>, doi:10.1007/8415\_2012\_125.
- [15] Felkel, P., Wegenkittl, R., Buhler, K., 2004. Surface models of tube trees, in: *Proceedings of the Computer Graphics International, IEEE Computer Society, Washington, DC, USA*, pp. 70–77. doi:10.1109/CGI.2004.93.
- [16] Flasque, N., Desvignes, M., Constans, J.M., Revenu, M., 2001. Acquisition, segmentation and tracking of the cerebral vascular tree on 3D magnetic resonance angiography images. *Medical Image Analysis* 5, 173–183.
- [17] Friman, O., Hindennach, M., Kühnel, C., Peitgen, H.O., 2010. Multiple hypothesis template tracking of small 3D vessel structures. *MedIA* 14, 160 – 171.
- [18] Gascuel, M.D., 1993. An implicit formulation for precise contact modeling between flexible solids, in: *Proc. of the 20th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1993*, pp. 313–320.

- [19] Goldman, R., 2005. Curvature formulas for implicit curves and surfaces. *Comput. Aided Geom. Des.* 22, 632–658. doi:10.1016/j.cagd.2005.06.005.
- [20] Gourmel, O., Barthe, L., Cani, M.P., Wyvill, B., Bernhard, A., Paulin, M., Grasberger, H., 2012. A gradient-based implicit blend. *ACM Trans. Graph.* .
- [21] Hahn, H.K., Preim, B., Selle, D., Otto Peitgen, H., 2001. Visualization and interaction techniques for the exploration of vascular structures, in: *Visualization*, 2001. VIS '01, pp. 395–402.
- [22] Hong, Q., Li, Q., Tian, J., 2012. Implicit reconstruction of vasculatures using bivariate piecewise algebraic splines. *IEEE Trans. Med. Imaging* 31, 543–553. doi:10.1109/TMI.2011.2172455.
- [23] Hoyos, M., Orłowski, P., Piatkowska-Janko, E., Bogrodzki, P., Orkisz, M., 2006. Vascular centerline extraction in 3D MR angiograms for phase contrast mri blood flow measurement. *International Journal of Computer Assisted Radiology and Surgery* 1, 51–61. doi:10.1007/s11548-006-0005-0.
- [24] Kazhdan, M., Bolitho, M., Hoppe, H., 2006. Poisson surface reconstruction, in: *Proceedings of the fourth Eurographics symposium on Geometry processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, pp. 61–70.
- [25] Kockara, S., Halic, T., Iqbal, K., Bayrak, C., Rowe, R., 2007. Collision detection: A survey, in: *Systems, Man and Cybernetics*, 2007. ISIC. IEEE International Conference on, pp. 4046–4051. doi:10.1109/ICSMC.2007.4414258.
- [26] Kretschmer, J., Godenschwager, C., Preim, B., Stamminger, M., 2013. Interactive patient-specific vascular modeling with sweep surfaces. *Visualization and Computer Graphics*, IEEE Transactions on 19, 2828–2837. doi:10.1109/TVCG.2013.169.
- [27] La Cruz, A., Straka, M., Kochl, A., Sramek, M., Groller, E., Fleischmann, D., 2004. Non-linear model fitting to parameterize diseased blood vessels, in: *Proceedings of the conference on Visualization '04*, IEEE Computer Society, Washington, DC, USA, pp. 393–400. doi:10.1109/VISUAL.2004.72.
- [28] Lachaud, J.O., Montanvert, A., 1999. Deformable meshes with automated topology changes for coarse-to-fine 3D surface extraction. *MedIA* 3, 187–207. doi:10.1016/S1361-8415(99)80006-1.
- [29] Lauric, A., Miller, E., Frisken, S., Malek, A., 2010. Automated detection of intracranial aneurysms based on parent vessel 3D analysis. *Medical Image Analysis* , 149–159doi:10.1016/j.media.2009.10.005.
- [30] Lauterbach, C., Yoon, S.E., Tuft, D., Manocha, D., 2006. RT-DEFORM: Interactive ray tracing of dynamic scenes using BVHs, in: *Interactive Ray Tracing 2006*, IEEE Symposium on, pp. 39–46. doi:10.1109/RT.2006.280213.
- [31] Lempitsky, V., 2010. Surface extraction from binary volumes with higher-order smoothness, in: *CVPR 2010*, pp. 1197–1204.
- [32] Lesage, D., Angelini, E., Bloch, I., et al., 2009. A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *MedIA* 13, 819–845. doi:http://dx.doi.org/10.1016/j.media.2009.07.011.
- [33] Li, H., Yezzi, A., 2006. Vessels as 4D curves: Global minimal 4D paths to extract 3D tubular surfaces, in: *Computer Vision and Pattern Recognition Workshop*, 2006. CVPRW '06. Conference on, pp. 82–82. doi:10.1109/CVPRW.2006.210.
- [34] Lin, M., Gottschalk, S., 1998. Collision detection between geometric models: A survey, in: *Proceedings of IMA conference on mathematics of surfaces*, pp. 602–608.
- [35] Luboz, V., Kyaw-Tun, J., Sen, S., Kneebone, R., Dickinson, R., Kitney, R., Bello, F., 2013. Real-time stent and balloon simulation for stenosis treatment. *The Visual Computer* , 1–9doi:10.1007/s00371-013-0859-4.
- [36] Mille, J., Cohen, L., 2009. Deformable tree models for 2D and 3D branching structures extraction, in: *Computer Vision and Pattern Recognition Workshops*, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on, pp. 149–156. doi:10.1109/CVPRW.2009.5204049.
- [37] Mohan, V., Sundaramoorthi, G., Tannenbaum, A., 2010. Tubular surface segmentation for extracting anatomical structures from medical imagery. *IEEE Trans. Med. Imaging* 29, 1945–1958. doi:10.1109/TMI.2010.2050896.
- [38] Mullen, P., de Goes, F., Desbrun, M., Cohen-Steiner, D., Alliez, P., 2010. Signing the unsigned: Robust surface reconstruction from raw pointsets, in: *Sorkine, O., Lévy, B. (Eds.), EUROGRAPHICS Symposium on Geometry Processing*, pp. 1733–1741.
- [39] Muraki, S., 1991. Volumetric shape description of range data using blobby model. *SIGGRAPH Comput. Graph.* 25, 227–235. doi:10.1145/122718.122743.
- [40] Oeltze, S., Preim, B., 2005. Visualization of vasculature with convolution surfaces: Method, validation and evaluation. *IEEE Trans. Med. Imaging* 24, 540–548. doi:http://dx.doi.org/10.1109/TMI.2004.843196.
- [41] Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.P., 2003. Multi-level partition of unity implicits, in: *ACM SIGGRAPH 2003 Papers*, ACM, New York, NY, USA, pp. 463–470. doi:10.1145/1201775.882293.
- [42] Preim, B., Oeltze, S., 2008. 3D visualization of vasculature: An overview, in: *Linsen, L., Hagen, H., Hamann, B., Farin, G., Hege, H.C., Hoffman, D., Johnson, C.R., Polthier, K., Rumpf, M. (Eds.), Visualization in Medicine and Life Sciences*. Springer Berlin Heidelberg. Mathematics and Visualization, pp. 39–59.
- [43] Schaap, M., Metz, C., van Walsum, T., van der Giessen, A., Weustink, A., Mollet, N., Bauer, C., Bogunovic, H., Castro, C., Deng, X., Dikici, E., O'Donnell, T., Frenay, M., Friman, O., Hoyos, M.H., Kitslaar, P., Krissian, K., KÄijhnel, C., Luengo-Oroz, M.A., Orkisz, M., Smedby, Ö., Styner, M., Szymczak, A., Tek, H., Wang, C., Warfield, S.K., Zambal, S., Zhang, Y., Krestin, G.P., Niessen, W., 2009a. Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms. *MedIA* 13/5, 701–714. doi:10.1016/j.media.2009.06.003.
- [44] Schaap, M., Neefjes, L., Metz, C., van der Giessen, A., Weustink, A., Mollet, N., Wentzel, J., van Walsum, T., Niessen, W., 2009b. Coronary lumen segmentation using graph cuts and robust kernel regression, in: *Information Processing and Medical Imaging (IPMI'09)*, pp. 528–539.
- [45] Schaap, M., Smal, I., Metz, C., Walsum, T., Niessen, W., 2007. Bayesian tracking of elongated structures in 3D images, in: *Karssemeijer, N., Lelieveldt, B. (Eds.), Information Processing in Medical Imaging*. Springer Berlin Heidelberg. volume 4584 of *Lecture*

- Notes in Computer Science*, pp. 74–85. doi:10.1007/978-3-540-73273-0\_7.
- [46] Schumann, C., Neugebauer, M., Bade, R., et al., 2008. Implicit vessel surface reconstruction for visualization and CFD simulation. *IJCARS* 2, 275–286. doi:<http://dx.doi.org/10.1007/s11548-007-0137-x>.
  - [47] Sherstyuk, A., 1999. Kernel functions in convolution surfaces: A comparative analysis. *The Visual Computer* 15, 171–182. doi:<http://dx.doi.org/10.1007/s003710050170>.
  - [48] Sierra, R., Bajka, M., Székely, G., 2003. Pathology growth model based on particles, in: *Proc. of Medical Image Computing and Computer Aided Intervention (MICCAI 2003)*, pp. 25–32. doi:10.1109/CVPR.2010.5539832.
  - [49] Tang, W., Wan, T.R., Gould, D.A., How, T., John, N.W., 2012. A stable and real-time nonlinear elastic approach to simulating guidewire and catheter insertions based on cosserat rod. *IEEE Trans. Biomed. Engineering* 59, 2211–2218. URL: <http://dx.doi.org/10.1109/TBME.2012.2199319>, doi:10.1109/TBME.2012.2199319.
  - [50] Taubin, G., 1991. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *PAMI* 13, 1115–1138. doi:10.1109/34.103273.
  - [51] Teschner, M., Kimmerle, S., Heidelberger, B., Zachmann, G., Raghupathi, L., Fuhrman, A., Cani, M.P., Faure, F., Magnenat-Thalmann, N., Strasser, W., Volino, P., 2005. Collision detection for deformable objects. *Computer Graphics Forum* 24, 61–81. doi:10.1111/j.1467-8659.2005.00829.x.
  - [52] Tsingos, N., Bittar, E., Cani, M.P., 1995. Implicit surfaces for semi-automatic medical organ reconstruction, in: *Computer Graphics Internat. (CGI'95)*, pp. 3–15.
  - [53] Tyrrell, J., di Tomaso, E., Fuja, D., et al., 2007. Robust 3-D modeling of vasculature imagery using superellipsoids. *IEEE Trans. Med. Imag.* 26, 223–237. doi:10.1109/TMI.2006.889722.
  - [54] Wang, Y., Narayanaswamy, A., Roysam, B., 2011. Novel 4-D open-curve active contour and curve completion approach for automated tree structure extraction, in: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1105–1112. doi:10.1109/CVPR.2011.5995620.
  - [55] Wink, O., Niessen, W.J., Viergever, M.A., 2000. Fast delineation and visualization of vessels in 3D angiographic images. *IEEE Trans. Med. Imaging* 19, 337–346. doi:<http://dx.doi.org/10.1109/42.848184>.
  - [56] Witkin, A.P., Heckbert, P.S., 1994. Using particles to sample and control implicit surfaces, in: *SIGGRAPH*, pp. 269–277.
  - [57] Wong, W., Chung, A., 2007. Probabilistic vessel axis tracing and its application to vessel segmentation with stream surfaces and minimum cost paths. *Medical Image Analysis* 11, 567–587. doi:<http://dx.doi.org/10.1016/j.media.2007.05.003>.
  - [58] Worz, S., Rohr, K., 2007. Segmentation and quantification of human vessels using a 3-D cylindrical intensity model. *Trans. Img. Proc.* 16, 1994–2004. doi:10.1109/TIP.2007.901204.
  - [59] Wu, J., Hu, Q., Ma, X., 2013. Comparative study of surface modeling methods for vascular structures. *Comp. Med. Imag. and Graph.* 37, 4–14.
  - [60] Wu, X., Luboz, V., Krissian, K., Cotin, S., Dawson, S., 2011. Segmentation and reconstruction of vascular structures for 3d real-time simulation. *MedIA* 15, 22 – 34. doi:10.1016/j.media.2010.06.006.
  - [61] Yim, P.J., Cebal, J.J., Mullick, R., B., M.H., Choyke, P.L., 2001. Vessel surface reconstruction with tubular deformable model. *IEEE Trans. Med. Imag.* 20, 1411–1421.